DOCUMENT RESUME

ED 248 860 IR 011 309

AUTHOR	Taitt, Kathy
TITLE	TLC for Growing Minds. APPLE II Programming Manual, Volume 4.
REPORT NO	ISBN-0-88193-004-0
PUB DATE	83
NOTE	59p.; For related documents, see IR 011 306-311, IR 011 314-316, and IR 011 318-319. This manual is also available for TRS-80, IBM PC, Commodore 64, PET, ATARI, TI-99/4A, and VIC-20.
AVAILABLE FROM	Creative Learning Association, Inc., RR#4, Box 330, Charleston, IL 61920 ($9.95 per copy; quantity discounts available).
PUB TYPE	Guides - Classroom Use - Materials (For Learner) (051)
EDRS PRICE	MF01 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS	Adult Education; *Computer Science Education; Computer Software; Elementary Secondary Education; Input Output Devices; Instructional Materials; *Learning Activities; *Microcomputers; Problem Solving; *Programing; Programing Languages
IDENTIFIERS	Apple II; *BASIC Programing Language

ABSTRACT

Designed to improve students' thinking, learning, and creative skills while they learn to program a microcomputer in BASIC programing language, this instructional manual for intermediate learners provides a brief section of information for each new topic together with a sample program to enter, run on the computer, and observe; a suggested change to clarify the principle being taught, and a challenge requiring use of the new principles. Each chapter includes a review. Individual chapters describe the subroutine commands GOSUB, RETURN, ON-GOTO, and ON-GOSUB; tabulating with the commands PRINT TAB(X), HTAB(X), and VTAB(X); SOUND, DIM(X), and CLEAR; and READ, DATA, RESTORE, TRACE, and NO TRACE. A reference chart displays commands and their functions. Separate color coded project books for elementary, junior high, and high school/adult students are available for this manual. (Author/LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

THINKING · LEARNING · CREATING

ED248860

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☑ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY

Marilyn Bufton

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR 011 309

THINKING · LEARNING · CREATING

TLC

APPLE II®
PROGRAMMING MANUAL

for Growing Minds

Volume 4

by Kathy Taitt

tion Resources

CREATIVE LEARNING ASSOCIATION, INC.
RR#4, Box 330
Charleston, Illinois 61920

ISBN 0-88193-004-0

© 1983 CREATIVE LEARNING ASSOCIATION, INC.

Charleston, Illinois 61920

All rights reserved. No part of this book, except for brief passages in articles and reviews that refer to both author and publisher, may be reproduced without written permission of the publisher.

Table of Contents

CHAPTER 1: LEARNING TO USE SUBROUTINES Page

GOSUB 1
RETURN 1
ON-GOTO 9
ON-GOSUB 13
Review 16

CHAPTER 2: TABULATING WITH PRINT TAB(X), HTAB(X), AND VTAB(X)

PRINT TAB(X) 17
HTAB(X) 21
VTAB(X) 21
Review 26

CHAPTER 3: SOUND, DIM(X), AND CLEAR

SOUND 27
DIM(X) 33
CLEAR 36
Review 39

CHAPTER 4: READ, DATA, RESTORE, AND TRACING

DATA 40
READ 40
RESTORE 44
TRACE 49
NO TRACE 49
Review 50
Reference 51

Creative Learning Association
Charleston, IL 61920

CHAPTER 1: LEARNING TO USE SUBROUTINES

INFORMATION

Lines 1ØØ and 11Ø in the program below are called a subroutine. A subroutine is a set of commands within a program. GOSUB tells the computer to GO to the SUBroutine. GOSUB must be followed by a line number.

The command RETURN must be at the end of a subroutine. RETURN sends the computer back to the line following the GOSUB command, which in this case is line 5Ø.

ENTER/RUN/OBSERVE

```
 1Ø HOME
 2Ø FOR X=1 TO 1Ø
 3Ø PRINT X:PRINT
 4Ø GOSUB 1ØØ
 5Ø NEXT X
 6Ø END
1ØØ FOR T=1 TO 1ØØØ:NEXT T
11Ø RETURN
```

CHANGE IT/RUN/OBSERVE

To see the difference the subroutine makes, delete line 4Ø and run the program again.

CHANGE IT/RUN/OBSERVE

Now put line 4Ø back into the program, and delete line 11Ø. No error message appears, but the program does not RETURN from the subroutine.

CHANGE IT/RUN/OBSERVE

Put line 11Ø back and this time, delete line 6Ø. This will cause the computer to find an extra RETURN command. RUN the program and notice the error message. "RETURN WITHOUT GOSUB" means the computer found an extra RETURN command. GOSUB and RETURN must be used in pairs like FOR and NEXT.

CHANGE IT/RUN/OBSERVE

Adjust your program so that it looks like this:

```
1Ø HOME
2Ø FOR X=1 TO 1Ø
3Ø PRINT X:PRINT
(program continued)
```

```
 40 GOSUB 100
 50 NEXT X
 60 GOTO 120
100 FOR T=1 TO 1000:NEXT T
110 RETURN
120 END
```

Lines 60 and 120 show another way to avoid the RETURN command when you are through with the subroutine.

CHANGE IT/RUN/OBSERVE

This will also work.

```
 60 GOTO 60
```

ENTER/RUN/OBSERVE

```
 10 HOME
 20 PRINT:PRINT:PRINT:PRINT
 30 PRINT "YOU ARE ABOUT TO TAKE THE SMITH
 QUIZ"
 40 PRINT:GOSUB 1000
 50 PRINT "EACH QUESTION MUST BE ANSWERED
 WITH A FIVE-LETTER NAME BEGINNING WITH
 S."
 60 GOSUB 1000
 70 HOME
 (program continued)
```

```
 80 PRINT "1.) A MAN WHO SHOES HORSES IS
 CALLED A BLACK_____."
 90 INPUT S$
100 GOSUB 2000
110 PRINT "2.) ADAM _____ WAS A FAMOUS
SCOTTISH ECONOMIST."
120 INPUT S$:GOSUB 2000
130 PRINT "3.) JOSEPH _____ WAS THE FOUNDER
OF THE MORMON CHURCH."
140 INPUT S$:GOSUB 2000
150 PRINT "4.) LOCATED IN WASHINGTON D.C.,
THE _____SONIAN INSTITUTION IS A WORLD-
FAMOUS MUSEUM."
160 INPUT S$:GOSUB 2000
170 PRINT "5.) A VIRGINIA COLONIST WAS
NAMED JOHN _____."
180 INPUT S$:GOSUB 2000
190 PRINT "THAT'S IT!  THE QUIZ IS OVER!"
200 PRINT:PRINT "YOU ANSWERED ";C;" OUT OF
5 CORRECTLY."
210 END
1000 REM *** TIME DELAY SUBROUTINE ***
1010 FOR T=1 TO 3000:NEXT T
1020 RETURN
2000 REM *** ANSWER SUBROUTINE ***
2010 HOME:PRINT
2020 IF S$<>"SMITH" THEN PRINT "NOPE!  THE
ANSWER IS SMITH.":GOTO 2050
2030 PRINT "RIGHT!  YOU ARE SO CLEVER."
2040 C=C+1
2050 GOSUB 1000
2060 HOME
```

(program continued)

```
 2070 RETURN
```

CHANGE IT/RUN/OBSERVE

Add extra spaces in those PRINT lines that need it to prevent a word being divided into two parts.

Notice, you can go to a subroutine from anywhere in a program. Also, you can have subroutines nested inside one another, in the same way that you did with FOR-NEXT commands. For example, in the answer subroutine, the computer branches to the time delay routine.

CHANGE IT/RUN/OBSERVE

Make these changes. LIST, then RUN.

```
   90 GOSUB 2000
  100
  120 GOSUB 2000
  140 GOSUB 2000
  160 GOSUB 2000
  180 GOSUB 2000
 2010 PRINT:INPUT S$
```

CHALLENGE

Change the Smith program so that two time delays are used, one so the instructions can be read, and a shorter one in the 2ØØØ subroutine.

ENTER/RUN/OBSERVE

```
 1Ø HOME:GR
 2Ø C=1:Z=2
 3Ø COLOR=C
 4Ø FOR N=Ø TO 38 STEP Z
 5Ø GOSUB 1ØØ
 6Ø GOSUB 2ØØ
 7Ø NEXT N
 8Ø END
1ØØ REM *** VERTICAL LINE SUBROUTINE ***
11Ø VLIN Ø,38 AT N
13Ø RETURN
2ØØ REM *** HORIZONTAL LINE SUBROUTINE ***
21Ø HLIN Ø,38 AT N
23Ø RETURN
```

CHANGE IT/RUN/OBSERVE

Add these lines:

```
120 GOSUB 300
220 GOSUB 300
300 FOR T=1 TO 200:NEXT T:RETURN
```

Can you predict what will happen before you run the program with these new lines added?

CHANGE IT/RUN/OBSERVE

Make these changes. LIST, then RUN.

```
  5 INPUT Z
 20 FOR C=1 TO 14
 25 X=0:Y=0
 75 NEXT C
 80 GOTO 5
120
220
```

CHANGE IT/RUN/OBSERVE

Try this:

```
 40 FOR N=10 TO 20 STEP 2
```

Try several numbers as INPUT. Make additional changes in the program if you wish.

CHALLENGE

Using subroutines, create a program that will draw a cage with bars. Make each bar a different color.

SAVE your program on tape or disk. Later you may want to put an animal in the cage.

ENTER/RUN/OBSERVE

```
 10 HOME
 20 A=1
 30 PRINT A
 40 IF A=1 GOTO 90
 50 IF A=2 GOTO 100
 60 IF A=3 GOTO 110
 70 IF A=4 GOTO 120
 80 IF A=5 GOTO 130
 90 PRINT " ONE":GOTO 140
100 PRINT " TWO":GOTO 140
110 PRINT " THREE":GOTO 140
120 PRINT " FOUR":GOTO 140
130 PRINT " FIVE"
140 A=A+1
 (program continued)
```

```
150 IF A=6 GOTO 150
160 GOTO 30
```

INFORMATION

The IF-THEN-GOTO lines 40 through 80 above can all be replaced with one ON-GOTO command.

CHANGE IT/RUN/OBSERVE

Delete lines 40, 50, 60, 70, and 80.

Now add a new line 40.

```
40 ON A GOTO 90,100,110,120,130
```

RUN again.

INFORMATION

Between the ON and GOTO, you must have either a variable or a mathematical expression that represents a number. The line numbers after GOTO tell the computer where to go. When the value of the variable is 1, the computer branches to the first line number listed. If the value is four, the computer will branch to the fourth

number listed. The value should be a non-zero, positive integer. What happens when this is not the case? Enter these next several examples to find out.

CHANGE IT/RUN/OBSERVE

```
**************************************************
* 2Ø A=Ø                                         *
**************************************************
```

Whenever the value is zero or greater than the number of line numbers to branch to, the computer just ignores the ON-GOTO command and goes on to the next line. Add:

```
**************************************************
* 5Ø PRINT "ZAP"                                 *
**************************************************
```

ZAP will only be printed when the computer does not branch to any of the lines listed in ON-GOTO.

CHANGE IT/RUN/OBSERVE

```
20 A=.7
```

When the expression is a decimal number, the ON-GOTO turns it into an integer by dropping the part to the right of the decimal point. All numbers are rounded down to the next lowest integer.

When you run the program, press CTRL C while the first line is still on the screen. Observe what happens.

ENTER/RUN/OBSERVE

```
10 HOME:PRINT:PRINT
20 PRINT "BECAUSE OF YOUR OUTSTANDING
QUALIFICATIONS, YOU HAVE JUST BEEN HIRED BY
XCORP, AN INTERNATIONAL CORPORATION."
30 PRINT:PRINT:PRINT
40 PRINT "TO FIND OUT YOUR PARTICULAR
POSITION IN THE COMPANY AND WHERE YOU WILL
BE LOCATED, ENTER A NUMBER FROM 1-4."
50 INPUT A
60 ON A GOTO 70,90,110,130
(program continued)
```

```
 70 PRINT:PRINT "TOO BAD.  YOU ARE A NEW
 EXECUTIVE IN THE PERFUME DIVISION, AND WILL
 BE LIVING IN PARIS, FRANCE.":GOSUB 500

 80 GOTO 550

 90 PRINT:PRINT "CONGRATULATIONS!  AS PART
 OF A PUBLIC RELATIONS PROJECT, YOU HAVE
 BEEN CHOSEN TO HEAD AN EXPEDITION TO
 ANTARCTICA TO STUDY THE SLEEPING HABITS OF
 PENGUINS.  BE SURE TO TAKE PLENTY OF WARM
 CLOTHING.":GOSUB 500

100 GOTO 550

110 PRINT:PRINT "THE GOOD NEWS IS THAT YOU
WILL BE WORKING IN NEW YORK CITY.  THE BAD
NEWS IS THAT YOUR JOB CONSISTS OF ARRANGING
THE REFRESHMENTS FOR OFFICE PARTIES.  LOOK
AT THE BRIGHT SIDE -- THE ONLY WAY TO GO IS
UP.":GOSUB 500

120 GOTO 550

130 PRINT:PRINT "SURPRISE!  THE COMPANY HAS
DECIDED TO MODERNIZE PRODUCTION OPERATIONS,
AND YOU ARE BEING SENT TO TOKYO, JAPAN, FOR
SIX MONTHS TO STUDY ROBOTICS.":GOSUB 500

140 GOTO 550

500 FOR T=1 TO 3000:NEXT T

510 PRINT:PRINT

520 RETURN

550 INPUT "DO YOU WANT TO GO AGAIN (Y/N) ";A$

560 IF A$="Y" GOTO 10

570 END
```

CHALLENGE

Create your own program using ON-GOTO.

Creative Learning Association
Charleston, IL 61920

INFORMATION

ON-GOSUB is the same as ON-GOTO except that the line numbers listed after ON-GOSUB are all the first lines of subroutines.

ENTER/RUN/OBSERVE

```
 10 HOME:PRINT:PRINT
 20 INPUT "WHAT IS YOUR NAME?";N$
 30 PRINT:PRINT "WHAT SPEED WOULD YOU LIKE?"
 40 PRINT "1=SLOW, 2=MEDIUM, AND 3=FAST"
 50 PRINT:INPUT "1, 2, OR 3?";S
 60 ON S GOSUB 100,120,140
 70 HOME:FOR H=1 TO 5
 80 PRINT N$:NEXT H
 90 END
100 SPEED=0
110 RETURN
120 SPEED=125
130 RETURN
140 SPEED=255
150 RETURN
```

RUN this program several times.

ENTER/RUN/OBSERVE

Here is an example of the kinds of programs you can create using both ON-GOSUB and ON-GOTO.

```
 10 HOME:PRINT:PRINT
 20 PRINT "THIS IS A QUIZ ON CAPITALS."
 30 GOSUB 330
 40 PRINT:PRINT "DO YOU WANT A STATE (1)
OR A COUNTRY (2)?"
 50 INPUT "ENTER 1 OR 2 ";N
 60 ON N GOSUB 70,200
 70 REM *** STATE CAPITALS ***
 80 HOME:IF S>2 THEN S=0
 90 S=S+1
100 ON S GOTO 110,140,170
110 INPUT "WHAT IS THE CAPITAL OF TEXAS?";S$
120 A$="AUSTIN"
130 GOTO 360
140 INPUT "WHAT IS THE CAPITAL OF MAINE?";S$
150 A$="AUGUSTA"
160 GOTO 360
170 INPUT "WHAT IS THE CAPITAL OF MONTANA?";
S$
180 A$="HELENA"
190 GOTO 360
200 REM *** COUNTRY CAPITALS ***
210 HOME:IF S>2 THEN S=0
220 C=C+1
230 ON C GOTO 240,270,300
240 INPUT "WHAT IS THE CAPITAL OF CANADA?";S$
250 A$="OTTAWA"
260 GOTO 360
270 INPUT "WHAT IS THE CAPITAL OF CHINA?";S$
(program continued)
```

```
280 A$="PEKING"
290 GOTO 360
300 INPUT "WHAT IS THE CAPITAL OF ISRAEL?";S$
310 A$="JERUSALEM"
320 GOTO 360
330 REM *** TIME DELAY ***
340 FOR T=1 TO 2000:NEXT T
350 RETURN
360 REM *** ANSWERS ***
370 PRINT:PRINT
380 IF S$<>A$ THEN PRINT "NOPE!  ";A$;" IS THE
CORRECT ANSWER.":GOTO 400
390 PRINT "RIGHT!!!"
400 GOSUB 330
410 PRINT:PRINT
420 INPUT "DO YOU WANT TO GO AGAIN (Y/N) ";V$
430 IF V$="Y" THEN HOME:GOTO 40
```

When you run this program, try the states and countries at least three times each.

DO NOT ERASE THIS PROGRAM!!!

CHALLENGE

Add two state capitals and two country capitals to the last program. Run the new program to be sure it works.

REVIEW: Choose the command or key which will cause the function described below.

COMMAND/KEY	FUNCTION
	This command sends the computer to the first line of a sub-routine. The command must be followed by a line number.
	Located at the end of a sub-routine, this command sends the computer back to the program line following GOSUB.
	This command sends the compu-ter to one of several line numbers, depending on the value of the expression in the command.
	This will send the computer to one of several subroutines, depending on the value of the expression in the command.

CHAPTER 2: TABULATING WITH PRINT TAB(X), HTAB(X), AND VTAB(X)

INFORMATION

On a typewriter, you use the TAB key to space over to a place on the page. The command to make your computer do this is PRINT TAB(X). X must be a number between Ø and 255.

ENTER/RUN/OBSERVE

```
1Ø HOME
2Ø PRINT "APPLE"
3Ø PRINT TAB(5) "PEAR"
4Ø PRINT TAB(1Ø) "ORANGE"
5Ø PRINT TAB(15) "BANANA"
```

CHANGE IT/RUN/OBSERVE

```
3Ø PRINT "PEAR" TAB(1Ø) "GRAPE"
```

Observe that the number X represents a position on the screen. PRINT TAB(X) means space to position X on the screen. What happens if the cursor is already past position X? Make the next change to find out!

CHANGE IT/RUN/OBSERVE

```
30 PRINT "PEAR" TAB(10) "GRAPE" TAB(5)
"KUMQUAT"
```

If the cursor is already past position X when it gets to PRINT TAB(X), it ignores the TAB(X) and PRINTs where it is. This is what happens with TAB(5) "KUMQUAT" above. The cursor has already passed position 5, so KUMQUAT is printed immediately after GRAPE.

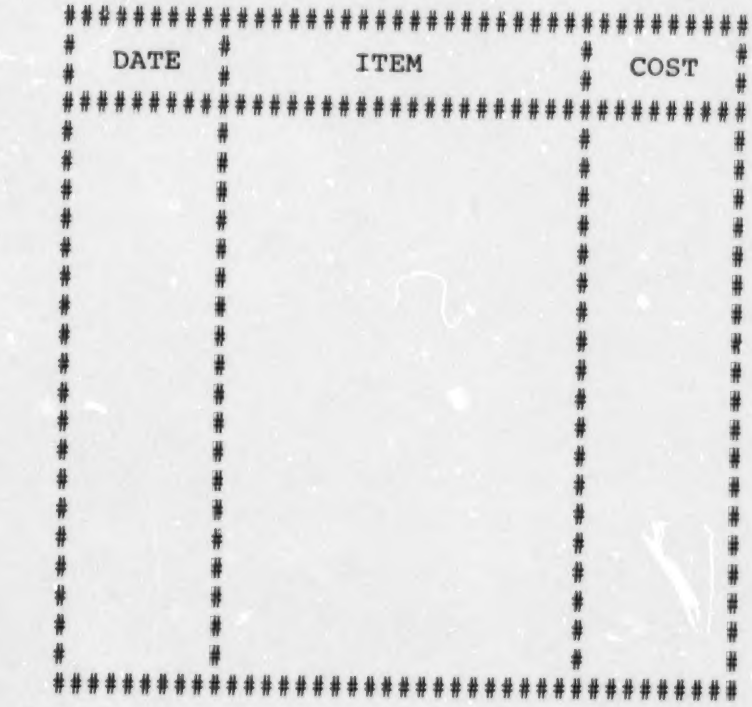
ENTER/RUN/OBSERVE

```
10 HOME
20 INPUT X
30 PRINT TAB(X) "APPLE"
40 GOTO 20
```

Use this program to experiment and discover more about PRINT TAB(X). It will help you to answer the CHALLENGE on the next page.

CHALLENGE

Using the PRINT TAB(N) command, create a program that will produce the form below.



ENTER/RUN/OBSERVE

```
10 HOME
20 FOR X=1 TO 10
30 Y=X+1
40 Z=X+Y
50 PRINT X TAB(8) Y TAB(16) Z
60 PRINT
70 NEXT X
```

PRINT TAB(X) is particularly useful for making tables.

CHALLENGE

For values of X from 1 to 8, make a table that looks like the following. Include labels over each column.

X	X*2	X*3	X*4	X*5
1	2	3	4	5
2	4	6	8	10
3	6	9	12	15

-etc.-

INFORMATION

HTAB and VTAB are somewhat similar to PRINT TAB(X), but there are several important differences.

HTAB moves the cursor horizontally. Like TAB(X), the values used with HTAB are 1 to 255. However, the value is not enclosed in parentheses. PRINT does not need to be used with HTAB.

The possible values for VTAB are 1 to 24. Parentheses are not used. VTAB moves the cursor to a particular line on the screen.

ENTER/RUN/OBSERVE

Compare this program to the one on page 17.

```
10 HOME
20 PRINT "APPLE"
30 PRINT TAB(10)"PEAR"
40 PRINT TAB(10)"ORANGE";
50 HTAB 5:PRINT "BANANA"
```

Notice that HTAB is a command by itself. You need to use a colon to include the PRINT command on the same line. With PRINT TAB(X), if the cursor is already past position X, the TAB command is ignored. This is not the case with HTAB or VTAB. In line 5Ø in the program for example, the cursor moves to the left to position 5 to print "BANANA."

ENTER/RUN/OBSERVE

```
10 HOME
20 INPUT X
30 HTAB X:PRINT "HTAB"
40 GOTO 20
```

Try lots of different X values to see how HTAB works.

CHANGE IT/RUN/OBSERVE

```
30 VTAB X:PRINT "VTAB"
```

CHANGE IT/RUN/OBSERVE

```
35 HTAB X:PRINT "HTAB"
```

CHALLENGE

Program your computer to create this display on the screen.

```
                    DAY
                  EACH
                UPWARD
              STEP
            ONE
       CLIMBING
      IS
SUCCESS
```

ENTER/RUN/OBSERVE

```
10 HOME
20 HTAB 10:VTAB 10
30 PRINT "PLEASE ENTER A LETTER."
40 HTAB 20:VTAB 12
50 INPUT A$
60 FOR N=1 TO 23
(program continued)
```

```
 70 VTAB N:PRINT A$:HTAB 40:PRINT A$;
 80 NEXT N
```

CHANGE IT/RUN/OBSERVE

Now add these lines.

```
 90 FOR P=1 TO 40
100 VTAB 1:HTAB P:PRINT A$;
110 VTAB 23:HTAB P:PRINT A$;
120 NEXT P
130 GOTO 130
```

ENTER/RUN/OBSERVE

Clear the last program from memory.

```
 10 HOME
 20 X=INT(RND(1)*40+1)
 30 Y=INT(RND(1)*23+1)
 40 HTAB X:VTAB Y:PRINT "HI!"
 50 GOTO 20
```

CHALLENGE

Change the program above so that snowflakes (*) instead of HIs are printed.

Creative Learning Association
Charleston, IL 61920

CHALLENGE

Create a program that will cause the word HOPE to start near the bottom left of the screen and move diagonally to the top right of the screen. Have it FLASH at each line as it moves upward. Use the SPEED command so it doesn't move too quickly.

```
                    HOPE
                  HOPE
                HOPE
              HOPE
            HOPE
          HOPE
        HOPE
      HOPE
    HOPE
  HOPE
HOPE
```

REVIEW: Choose the command or key which will cause the function described below.

COMMAND/KEY	FUNCTION
	This command moves the cursor to the right to position X. (X=1 to 255.)
	This command moves the cursor to the horizontal position X, which could be to the right or left. (X=1 to 255.)
	This command moves the cursor to line X on the screen. (X=1 to 24.)

CHAPTER 3: SOUND, DIM(X), AND CLEAR

INFORMATION

Your computer often makes a beeping noise when you are creating a program--whenever you make an error, for example.

ENTER/RUN/OBSERVE

```
10 PRINT CHR$(7)
20 GOTO 10
```

ENTER/RUN/OBSERVE

This is a way to produce a different sound with your computer.

```
10 S=PEEK(-16336)
20 GOTO 10
```

CHANGE IT/RUN/OBSERVE

Add these lines.

```
 5 FOR T=1 TO 10
20 NEXT T
```

ENTER/RUN/OBSERVE

Try this:

```
10 INPUT N
20 FOR T=1 TO 30
30 S=PEEK(-16336)
40 FOR X=1 TO N:NEXT X
50 NEXT T
60 GOTO 10
```

INPUT different values of N and hear what it sounds like.

CHALLENGE

Can you pick a value of N that sounds like:

a) a machine gun

b) a motorcycle

c) a clock ticking

d) a buzzer

ENTER/RUN/OBSERVE

```
30 GR:FOR X=1 TO 39
40 COLOR=2:PLOT X,4
50 GOSUB 100
60 COLOR=16:PLOT X-1,4
(program continued)
```

```
  70 NEXT X
  80 GOSUB 100
  90 GOSUB 200
  95 END
 100 FOR T=1 TO 10:NEXT T
 110 RETURN
 200 FOR D=1 TO 5
 210 S=PEEK(-16336)
 220 GOSUB 100
 230 NEXT D:RETURN
```

CHANGE IT/RUN/OBSERVE

To make changes easily, add these lines.

```
  10 INPUT B
  20 INPUT C
 100 FOR T=1 TO B:NEXT T
 200 FOR D=1 TO C
```

The value of B controls the speed of the blue blob. The value of C controls the sound.

ENTER/RUN/OBSERVE

```
  10 GR:Y=1:HOME
  20 GOSUB 100:GOSUB 200
  30 Y=Y+1:GOTO 20
 100 REM *** BLOB RIGHT ***
  (program continued)
```

```
 11Ø FOR X=1 TO 38
 12Ø COLOR=2:PLOT X,Y
 13Ø COLOR=16:PLOT X-1,Y
 14Ø NEXT X:GOSUB 3ØØ:RETURN
 2ØØ REM *** BLOB LEFT ***
 21Ø FOR X=38 TO 1 STEP -1
 22Ø COLOR=1:PLOT X,Y
 23Ø COLOR=16:PLOT X+1,Y
 24Ø NEXT X:GOSUB 3ØØ:RETURN
 3ØØ REM *** SOUND ***
 31Ø FOR D=1 TO 1Ø
 32Ø S=PEEK(-16336)
 33Ø NEXT D:RETURN
```

CHANGE IT/RUN/OBSERVE

```
 135 IF X=2Ø THEN PRINT CHR$(7)
```

CHANGE IT/RUN/OBSERVE

Make these changes.

```
 135
  2Ø GOSUB 1ØØ:Y=Y+1
  3Ø GOSUB 2ØØ:Y=Y+1:GOTO 2Ø
```

CHANGE IT/RUN/OBSERVE

To prevent the error message, add this.

```
205 IF Y>37 THEN END
```

CHALLENGE

a) Change the previous program so that the right wall appears to move one space to the left each time the blob hits it.

b) Change it so the right wall appears to move two spaces to the left each time the blob hits it.

c) Change it so BOTH walls appear to move one space towards each other each time either is hit by the blob. Have the program stop when the two walls come together.

ANOTHER CHALLENGE

Morse code is used to send messages. The Morse code represents every letter as a combination of dots and dashes. Dots are represented by a short beep, dashes by a long one.

Your challenge is to program the computer to say "HELLO" in Morse code.

THE INTERNATIONAL MORSE CODE

A= .-	J= .---	S= ...
B= -...	K= -.-	T= -
C= -.-.	L= .-..	U= ..-
D= -..	M= --	V= ...-
E= .	N= -.	W= .--
F= ..-.	O= ---	X= -..-
G= --.	P= .--.	Y= -.--
H=	Q= --.-	Z= --..
I= ..	R= .-.	

INFORMATION

It is possible to have up to 89 storage locations instead of 11 in an array when you use the DIM command.

The command DIM A$(2Ø) would create 21 locations for the array A$. You must include the name of the array and the size you wish it to be.

When an array is DIMensioned, memory space is set aside for it. For this reason, it is a good idea not to dimension arrays larger than you really need them to be because you will be wasting computer memory space.

ENTER/RUN/OBSERVE

```
1Ø DIM N(14)
3Ø HOME
4Ø PRINT "PLEASE ENTER 15 NUMBERS, ONE AT
A TIME."
5Ø FOR T=Ø TO 14
6Ø INPUT N(T)
9Ø NEXT T
(program continued)
```

```
100 HOME
110 PRINT "HERE ARE THE NUMBERS."
120 FOR X=0 TO 14
130 PRINT "N(";X;")=";N(X)
140 NEXT X
```

CHANGE IT/RUN/OBSERVE

Delete line 10 and RUN the program again.

ENTER/RUN/OBSERVE

```
 10 HOME:PRINT:PRINT
 20 PRINT "HOW MANY TRIPS WOULD YOU LIKE TO
TAKE?"
 30 INPUT G
 40 IF G>11 THEN DIM T$(G-1)
 50 HOME:PRINT " NAME THE PLACES YOU WANT TO
VISIT."
 60 FOR X=0 TO G-1
 70 INPUT T$(X)
 80 NEXT X
 90 HOME
100 PRINT "OKAY, HERE ARE YOUR DESTINATIONS
:":PRINT
110 FOR X=G-1 TO 0 STEP -1
120 PRINT T$(X)
130 NEXT X
140 PRINT:PRINT:PRINT " BON VOYAGE!"
```

Creative Learning Association
Charleston, IL 61920

CHALLENGE

Write a program that asks how many people you wish to invite to a party. Create an array for the desired guests and, one by one, put their names into the array. Finally, print out the guest list on the screen.

Try running the program one time with less than 12 guests and a second time with more than 12 guests.

Creative Learning Association
Charleston, IL 61920

INFORMATION

The command CLEAR sets the values of all variables, strings, and arrays to zero. It CLEARs any and all values the computer has assigned.

ENTER/RUN/OBSERVE

```
10 FOR Z=1 TO 5
20 X=X+1
30 PRINT X
40 NEXT Z
50 GOTO 10
```

CHANGE IT/RUN/OBSERVE

```
45 CLEAR
```

Compare the display of the original program to this new one to see the difference CLEAR makes.

ENTER/RUN/OBSERVE

```
10 HOME
20 FOR Z=1 TO 10
30 X=INT(RND(1)*101)
(program continued)
```

```
 40 PRINT X
 50 REM ** E=EVEN, O=ODD, S=SUM OF THE
NUMBERS, A=AVERAGE **
 60 IF X/2=INT(X/2) THEN E=E+1:GOTO 80
 70 O=O+1
 80 S=X+S
100 NEXT Z
110 PRINT:PRINT
120 PRINT E;" EVEN NUMBERS WERE CHOSEN."
130 PRINT O;" ODD NUMBERS WERE CHOSEN."
140 PRINT "THE AVERAGE OF THE NUMBERS IS ";
S/10
150 PRINT:PRINT "DO YOU WANT TO GO AGAIN?"
160 INPUT "Y OR N?";A$
170 IF A$="N" THEN END
180 CLEAR:GOTO 10
```

CHANGE IT/RUN/OBSERVE

Change line 180 to:

```
 180 GOTO 10
```

RUN the program a few times and look at the totals and average you get without CLEAR!

CHALLENGE

Create a subroutine that will store questions that you INPUT in one array. Create another subroutine that will store answers that you INPUT in a second array. Have a third subroutine that will randomly select and PRINT a question. Have a fourth subroutine that will randomly select and PRINT an answer. Make your questions and answers humorous. Can you add appropriate sound effects?

Creative Learning Association
Charleston, IL 61920

REVIEW: Choose the command which will cause the function described below.

COMMAND	FUNCTION
	This command causes the computer to make a beeping noise.
	Using this command allows you to create arrays with more than 11 storage locations.
	This command sets the values of all variables, strings, and arrays to Ø.

Creative Learning Association
Charleston, IL 61920

CHAPTER 4: READ, DATA, RESTORE, AND TRACING

INFORMATION

The commands DATA and READ allow you to assign values to a large number of variables quickly.

ENTER/RUN/OBSERVE

```
10 HOME
20 DATA 1,2,3,4
30 READ A,B,C,D
40 PRINT
50 PRINT A;B;B;C;C;C;D;D;D;D
```

Values in the DATA statement must be separated by commas. The same is true for the variables in the READ statement.

The READ command uses the values in the DATA statement one by one.

CHANGE IT/RUN/OBSERVE

DATA statements can be listed anywhere within a program.

Delete line 20.

```
60 DATA 1,2,3,4
```

CHANGE IT/RUN/OBSERVE

```
70 PRINT:PRINT A,B,C,D
```

CHANGE IT/RUN/OBSERVE

Here's another example of the difference CLEAR makes.

```
65 CLEAR
```

Now RUN again.

ENTER/RUN/OBSERVE

```
10 HOME
20 DATA ONE
30 DATA TWO, THREE, FOUR
40 FOR N=1 TO 6
50 READ X$
60 HTAB 18:VTAB N:PRINT X$
70 NEXT N
80 DATA FIVE, SIX
```

INFORMATION

In a READ statement, use string variables for strings and regular variables for numbers.

A READ command reads DATA values in the order they are listed in a program. The first DATA value is assigned to the first READ variable, the second DATA value to the second READ variable, and so on.

CHALLENGE

Predict what will be printed when the program listed below is RUN. Enter and RUN the program to see if you were right.

```
10 HOME
20 DATA ILLINOIS,1,2
30 READ A$
40 DATA NEW YORK CITY,CALIFORNIA
50 DATA LOS ANGELES,3,NEW YORK
60 READ X,Y,B$,C$,D$,Z
70 READ E$,F$
80 DATA CHICAGO
90 PRINT:PRINT X;" ";B$
100 PRINT Y;" ";F$
110 PRINT Z;" ";D$
```

ENTER/RUN/OBSERVE

```
10 DIM M$(12)
20 DATA JANUARY,FEBRUARY,MARCH,APRIL,MAY,
JUNE,JULY,AUGUST,SEPTEMBER,OCTOBER,NOVEMBER,
DECEMBER
30 FOR N=1 TO 12
40 READ M$(N)
50 NEXT N
60 HOME:PRINT:PRINT
70 PRINT TAB(3)"TYPE THE NUMBER OF THE MONTH
YOU WERE BORN"
80 INPUT N
90 PRINT:PRINT "THAT MEANS YOUR BIRTHDAY
IS IN ";M$(N);"!"
```

CHALLENGE

Create a program using DATA and READ that lets you INPUT a month and day of the week and the year. The computer will then PRINT the date numerically as MM/DD/YY. For example, June 1, 1983, would be printed as 06/01/83.

ERIC Clearinghouse on Information Resources
School of Education
Syracuse University
Syracuse, New York 13210
(315) 423-3640

INFORMATION

The command RESTORE makes the computer return to the first DATA value in the program.

ENTER/RUN/OBSERVE

```
10 DATA 157,223,700
20 READ A,B,C
30 READ D,E,F
40 PRINT A,B,C,D,E,F
```

Notice the error message you get when you try to RUN this program?

CHANGE IT/RUN/OBSERVE

```
25 RESTORE
```

ENTER/RUN/OBSERVE

```
10 HOME:PRINT
20 PRINT TAB(2) "HERE IS A LIST OF THE
PEOPLE I CAN BEAT IN TENNIS:":PRINT
30 DATA MY GRANDMOTHER,CHRIS EVERT-LLOYD
40 DATA TRACY AUSTIN,MARTINA NAVRATILOVA
45 DATA JOHN MCENROE,IVAN LENDL
50 FOR N=1 TO 7
(program continued)
```

```
  60 READ T$:PRINT T$
  70 RESTORE
  80 NEXT N
```

CHALLENGE

Create a program that asks you to INPUT your name. IF the name you INPUT is BJORN BORG, then the program will PRINT all the names in lines 30, 40, and 45 in the above program. If you INPUT JIMMY CONNORS, have it PRINT only the mens' names.

ENTER/RUN/OBSERVE

Here's a program that uses many of the new commands you have learned. Notice that three different ways are used to draw the graphic letters. You will be asked to SAVE this program when you have finished. So, if you are using disks, be sure to boot up a disk before you start.

```
  10 HOME:PRINT:PRINT
  20 PRINT:INPUT "PLEASE ENTER A NUMBER (1-4)
   ";N
  (program continued)
```

```
 50 ON N GOSUB 80,150,220,330
 60 FOR T=1 TO 2000:NEXT T
 70 TEXT:GOTO 10
 80 REM *** SUBROUTINE FOR T ***
 90 GR:COLOR=N
100 FOR X=2 TO 12
110 PLOT X,13:NEXT X
120 FOR Y=13 TO 24
130 PLOT 7,Y:NEXT Y
140 RETURN
150 REM *** SUBROUTINE FOR L ***
160 GR:COLOR=N
170 VLIN 13,24 AT 16
180 HLIN 16,23 AT 24
210 RETURN
220 REM *** SUBROUTINE FOR C ***
230 GR:COLOR=N
240 RESTORE
250 FOR Y=15 TO 22
260 PLOT 27,Y:NEXT Y
270 DATA 28,14,29,13,30,13,31,13,32,13,33,
14,34,15
280 DATA 28,23,29,24,30,24,31,24,31,24,32,
24,33,23,34,22
290 FOR Z=1 TO 15
300 READ X,Y:PLOT X,Y
310 NEXT Z:RETURN
330 REM *** SUBROUTINE FOR TLC ***
340 GR:COLOR=N
(program continued)
```

```
35Ø GOSUB 1ØØ:GOSUB 17Ø:GOSUB 24Ø
36Ø RETURN
```

RUN the program at least four times, trying all the options. Then delete line 24Ø and run through all four choices again. Do you see why the RESTORE command is necessary?

Replace line 24Ø.

CHANGE IT/RUN/OBSERVE

Let's make some changes.

```
4Ø INPUT "PLEASE ENTER A NUMBER (1-15)";N
5Ø ON N GOSUB 8Ø,15Ø,22Ø,33Ø,8Ø,15Ø,22Ø,
33Ø,33Ø,22Ø,15Ø,8Ø,33Ø,33Ø,33Ø
```

CHANGE IT/RUN/OBSERVE

Why not let the computer do the work?

```
4Ø FOR N=1 TO 15
7Ø NEXT N
```

CHALLENGE

Change the last program to make the computer print your favorite monogram on the screen. Use the method that you like best.

SAVE THIS PROGRAM!!

Creative Learning Association
Charleston, IL 61920

INFORMATION

The command TRACE makes the computer print the line number of each line as it is run. This can be helpful when trying to locate an error in a program. The line numbers are usually printed on the left side of the screen with a # sign before each one.

When the computer is in the GR mode, the line numbers will appear at the bottom of the screen.

The command NOTRACE turns this debugging command off.

ENTER/RUN/OBSERVE

With your program from the last challenge still in the computer, enter:

```
TRACE
```

Now RUN. Can you follow the program?

ENTER/RUN/OBSERVE

Now turn it off.

```
NOTRACE
```

REVIEW: Choose the command which will cause the function described below.

COMMAND	FUNCTION
	This command is used to put lists of data into a program.
	This command reads the data values and assigns variable names to them.
	This command resets the READ command to the first DATA statement in a program.
	Use this to turn the debug mode on. It displays each line number as it is executed.
	This turns the debug mode off.

REFERENCE

COMMAND/KEY	FUNCTION
CLEAR	This command sets the values of all variables, strings, and arrays to Ø.
DATA	This command is used to put lists of data into a program.
DIM(X)	Using this command allows you to create arrays with more than 11 storage locations.
GOSUB	This command sends the computer to the first line of a sub-routine. The command must be followed by a line number.
HTAB(X)	This command moves the cursor to the horizontal position X, which could be to the right or left. (X=1 to 255.)
NO TRACE	This turns the debug mode off.
ON-GOSUB	This will send the computer to one of several subroutines, depending on the value of the expression in the command.

REFERENCE (continued)

COMMAND/KEY	FUNCTION
ON-GOTO	This command sends the computer to one of several line numbers, depending on the value of the expression in the command.
PEEK(-16336)	This command causes the computer to make a beeping noise.
PRINT TAB(X)	This command moves the cursor to the right to position X. (X=1 to 255.)
READ	This command reads the data values and assigns variable names to them.
RESTORE	This command resets the READ command to the first DATA statement in a program.
RETURN	Located at the end of a subroutine, this command sends the computer back to the program line following GOSUB.
TRACE	Use this to turn the debug mode on. It displays each line number as it is executed.

REFERENCE (continued)

COMMAND/KEY	FUNCTION
VTAB(X)	This command moves the cursor to line X on the screen. (X=1 to 24.)

CONGRATULATIONS!!!

You have now finished the green instructional level of TLC FOR GROWING MINDS and have a good working vocabulary of BASIC commands.

You are now ready to work the GREEN projects in the TLC Project Book, Volume 3 & 4. They will help you gain experience using the commands you have already learned. When you have finished the GREEN projects, Volume 5 (BLUE) of TLC FOR GROWING MINDS will help you to continue learning.

Creative Learning Association
Charleston, IL 61920